PINN Update: Baseline Enhancement and Parametric Extension

Ali Haghighi

Supervised by:

Afshin Ashrafzadeh, François Lehmann, Marwan Fahs

January 19, 2026

Abstract

This report presents recent progress on baseline and parametric Physics-Informed Neural Networks (PINNs) for the one-dimensional advection–dispersion equation formulated in a fully dimensionless setting. The baseline PINN approximates the concentration C^* as a function of space and time (x^*, t^*), while the parametric extension augments the input space to $(x^*, t^*, \log Pe)$, enabling a single model to represent solutions across a wide range of Peclet numbers. Both scripts are designed for clarity, reproducibility, and ease of understanding, following the guidance emphasized by Dr. Lehmann to keep the implementations linear and easy to follow, with full code available at baseline PINN and parametric PINN.

In addition to the main baseline and parametric models, I also explored complementary experimental directions to assess robustness and potential performance improvements. Appendix A reports a grid search evaluating training efficiency and sensitivity to architectural and sampling choices, while Appendix B documents an exploratory adaptive collocation strategy combined with an Adam–LBFGS optimizer schedule.

1 Methods

1.1 Baseline PINN

The baseline model solves the dimensionless advection–dispersion equation with a fixed Pe:

$$\frac{\partial C^*}{\partial t^*} + \frac{\partial C^*}{\partial x^*} = \frac{1}{Pe}\frac{\partial^2 C^*}{\partial x^{*2}}$$

The dimensionless domains and conditions are:

$$\begin{aligned} x^* &\in [0,1] \\ t^* &\in [0,1] \\ C^*(x^*,0) &= 0 \\ C^*(0,t^*) &= 1 \\ C^*(1,t^*) &= 0 \end{aligned}$$

Guided by the grid-search results in Appendix A, the baseline PINN uses a compact fully connected architecture with 3 hidden layers of 12 neurons (Tanh) to approximate the mapping $(x^*, t^*) \mapsto C^*$, written in the same functional form used later for the parametric model but without the $\log Pe$ input. This configuration offered a favorable accuracy–runtime tradeoff in our tests. For reference, the 3×12 network has 361 trainable parameters, compared with 257 parameters for the 1×64 single-hidden-layer variant suggested during the meeting; despite being slightly larger, the deeper 3×12 configuration converged more reliably to substantially lower losses in the explored settings. Xavier normal initialization [1] is applied to all linear layers to keep forward activations and backward gradients at comparable scales across depth, reducing vanishing/exploding behavior and improving training stability. Training is implemented with a closure-style loss block and automatic differentiation to compute $C^*_{t^*}$, $C^*_{x^*}$, and $C^*_{x^*x^*}$. Collocation points are sampled uniformly over (x^*, t^*) using 300×300 interior points and 300 points each for the initial and boundary conditions. Optimization uses Adam ($\mathrm{lr} = 10^{-3}$) for 15,000 epochs with all loss-term weights set to 1.

1.2 Parametric PINN

The parametric model extends the baseline formulation by augmenting the input with $\log Pe$ and learning the mapping $(x^*, t^*, \log Pe) \mapsto C^*$ across a continuous range of Peclet numbers. This enables a single network to represent solutions spanning diffusion-dominated to advection-dominated regimes. In the current implementation, $Pe_{\min} = 1$ and $Pe_{\max} = 10^5$, with $x^* \in [0,1]$ and $t^* \in [0,1]$. The governing equation remains

$$\frac{\partial C^*}{\partial t^*} + \frac{\partial C^*}{\partial x^*} = \frac{1}{Pe}\frac{\partial^2 C^*}{\partial x^{*2}}$$

The dimensionless domains and conditions are

$$\begin{aligned} x^* &\in [0,1] \\ t^* &\in [0,1] \\ Pe &\in [Pe_{\min}, Pe_{\max}] \\ C^*(x^*,0) &= 0 \\ C^*(0,t^*) &= 1 \\ C^*(1,t^*) &= 0 \end{aligned}$$

1.2.1 Architecture and initialization

The parametric network is a fully connected feedforward model with 4 hidden layers and 16 neurons per layer, using Tanh activations. Inputs are $(x^*, t^*, \log Pe)$ and the scalar output is C^*. Xavier normal initialization with a Tanh gain is applied to all linear layers to stabilize activation and gradient magnitudes across depth, which is particularly important for the increased input dimensionality [1].

1.2.2 Collocation sampling

To avoid overfitting to a fixed collocation cloud, new collocation sets are sampled at each training epoch. Interior PDE points are drawn uniformly with $N_{\mathrm{coll}} = 250 \times 250$. Initial condition points are sampled at $t^* = 0$ with $N_{\mathrm{IC}} = 250$, and boundary condition points are sampled at $x^* = 0$ (inlet) and $x^* = 1$ (outlet) with $N_{\mathrm{BC}} = 250$ each. For all sets, x^* and t^* are sampled uniformly on $[0,1]$, while $\log Pe$ is sampled uniformly in $[\log Pe_{\min}, \log Pe_{\max}]$ so that each order of magnitude in Pe is equally represented during training.

1.2.3 Automatic differentiation and residual

The PDE residual is evaluated at the interior points using automatic differentiation:

$$R_{\mathrm{PDE}} = \frac{\partial C^*}{\partial t^*} + \frac{\partial C^*}{\partial x^*} - \frac{1}{Pe}\frac{\partial^2 C^*}{\partial x^{*2}}, \qquad Pe = \exp(\log Pe).$$

First derivatives are computed using `grad(..., create_graph=True)`, and the second derivative is obtained by differentiating $\partial C^*/\partial x^*$ with respect to x^*.

1.2.4 Loss function and training loop

The total loss is defined as an unweighted sum of the PDE residual, initial condition, and boundary condition losses:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{PDE}} + \mathcal{L}_{\text{IC}} + \mathcal{L}_{\text{inlet}} + \mathcal{L}_{\text{outlet}}.$$

Loss terms mirror those of the baseline model; the only difference is that the PDE residual is evaluated using the sampled $Pe = \exp(\log Pe)$ at each collocation point. Training uses AdamW with learning rate 10^{-3} and weight decay equal to lr/10 for 15,000 epochs. Loss histories are logged for diagnostics before generating C^* profiles for selected Peclet numbers and time slices.

1.2.5 Parametric application to the baseline case

To recover the baseline physical problem from the trained parametric model, physical variables are mapped to the dimensionless inputs required by the network and the output is rescaled to physical units. The physical advection–dispersion equation

$$\frac{\partial C}{\partial t} + U\frac{\partial C}{\partial x} = D\frac{\partial^2 C}{\partial x^2}$$

is nondimensionalized using

$$x^* = \frac{x}{L}, \quad t^* = \frac{t}{T}, \quad C^* = \frac{C}{C_0}, \quad T = \frac{L}{U},$$

which yields

$$\frac{\partial C^*}{\partial t^*} + \frac{\partial C^*}{\partial x^*} = \frac{1}{Pe}\frac{\partial^2 C^*}{\partial x^{*2}}, \qquad Pe = \frac{UL}{D}.$$

For the baseline parameters $U = 0.1$ m/day, $D = 1.0 \times 10^{-7} \times 86400$ m^2/day, $C_0 = 5$ kg/m^3, and $L = 100$ m, we obtain $Pe \approx 1157.41$ and set $\log Pe = \log(UL/D)$. For a given physical time t and spatial coordinate x, the network is evaluated at $(x^*, t^*, \log Pe)$ with $x^* = x/L$ and $t^* = t/(L/U)$, producing $C^*(x^*, t^*; \log Pe)$. The physical concentration is then recovered as

$$C(x, t) = C_0\, C^*(x^*, t^*; \log Pe).$$

1.2.6 Analytical overlay

The one-dimensional Ogata–Banks analytical solution [2] is overlaid as a qualitative reference rather than a training target. The closed-form solution is evaluated on the plotting grid at selected time slices and rendered as dashed curves alongside the PINN outputs. To remain in dimensionless form, the overlay is computed with $U = 1$, $C_0 = 1$, and $D = 1/Pe$, ensuring consistency with x^* and t^* while reflecting the intended Peclet regime.

2 Results

2.1 Baseline PINN

The baseline model is trained in dimensionless form on $C^*(x^*, t^*)$. Figure 1 reports the corresponding rescaled dimensional concentration C, using the same physical parameters as the baseline study: $U = 0.1$ m/day, $D = 1.0 \times 10^{-7} \times 86400$ m^2/day, $C_0 = 5$ kg/m^3, with $x \in [0, 100]$ m and $t \in [0, 1000]$ days. With Xavier initialization and the revised training procedure, the PINN predictions are now almost perfectly aligned with the analytical solution across the domain. The only visible discrepancy appears immediately after the sharp advective front, where the concentration briefly undershoots below zero before relaxing back to zero as the profile flattens. This localized artifact represents the most noticeable remaining mismatch; it could be mitigated in future work by enforcing physical constraints such as non-negativity of concentration and monotonicity in space. Overall, the current baseline model exhibits stable training behavior and highly accurate solution profiles.

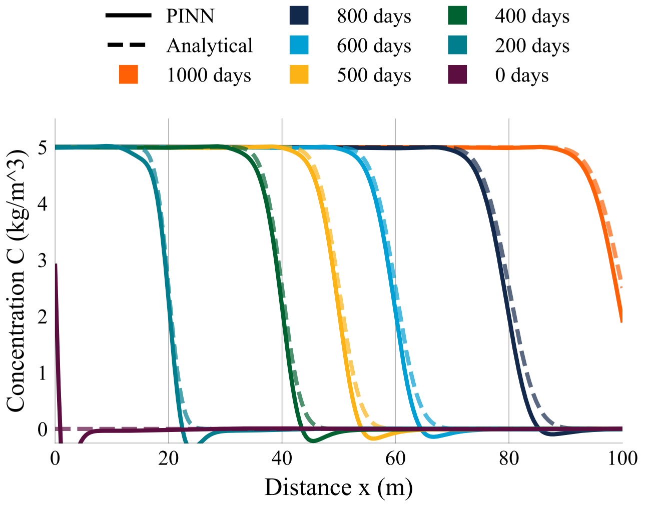


Figure 1: Baseline PINN concentration profiles C (kg/m^3) over x at selected t values with analytical overlay (shape reference, inlet $C_0 = 5$).

2.2 Parametric PINN application (physical units)

Figure 2 applies the trained parametric PINN to the baseline physical parameter set and reports the concentration $C(x, t)$ in physical units. The model is evaluated at the baseline Peclet number $Pe \approx 1157.41$, demonstrating that the parametric formulation can accurately recover the original single-Pe problem without retraining.

2.3 Parametric PINN

Figure 3 presents dimensionless concentration profiles C^* predicted by the parametric PINN for representative Peclet numbers $Pe = 10$, 50, 500, and 10^5, selected from the full training range $Pe \in [1, 10^5]$. Across all cases, the model reproduces the expected transition from smooth, diffusion-dominated fronts at low Pe to increasingly sharp, advection-dominated fronts at higher Pe, with good agreement with the analytical reference solutions.

For $Pe = 10$, the analytical solution has not yet fully relaxed to zero at the outlet for the larger plotted times (e.g., $t^* = 0.6$, 0.8, and 1.0), whereas the PINN predictions enforce the

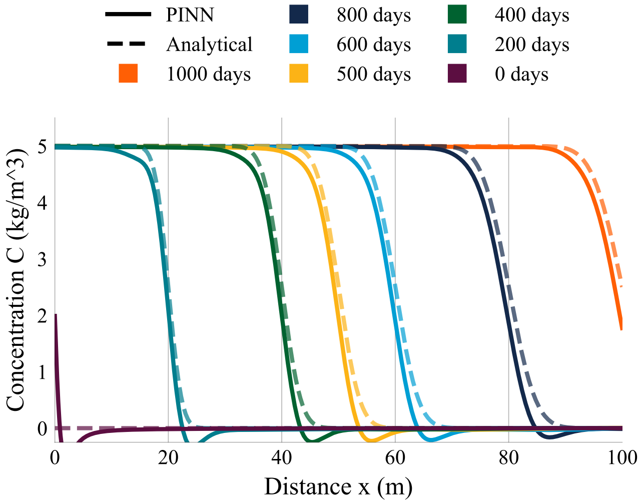


Figure 2: Parametric PINN applied to the baseline physical problem. The model is evaluated at the baseline $Pe \approx 1157.41$ and rescaled to C (kg/m^3) over x for selected times t.

outlet boundary condition $C^*(1, t^*) = 0$ as imposed during training. As a result, the predicted profiles deviate from the analytical curves near $x^* = 1$, decaying to zero earlier than the analytical solution, which naturally reaches zero only after $t^* > 1$. A similar, though weaker, effect is observed for $Pe = 50$ at later times, reflecting the same interaction between finite-time analytical behavior and the enforced boundary condition.

As in the baseline model, the most visible discrepancy occurs just behind the sharp advective front, especially at higher Peclet numbers, where the PINN briefly undershoots below zero before relaxing back to zero. This artifact remains localized. Overall the parametric PINN maintains stable training and high accuracy across the full Peclet range considered.

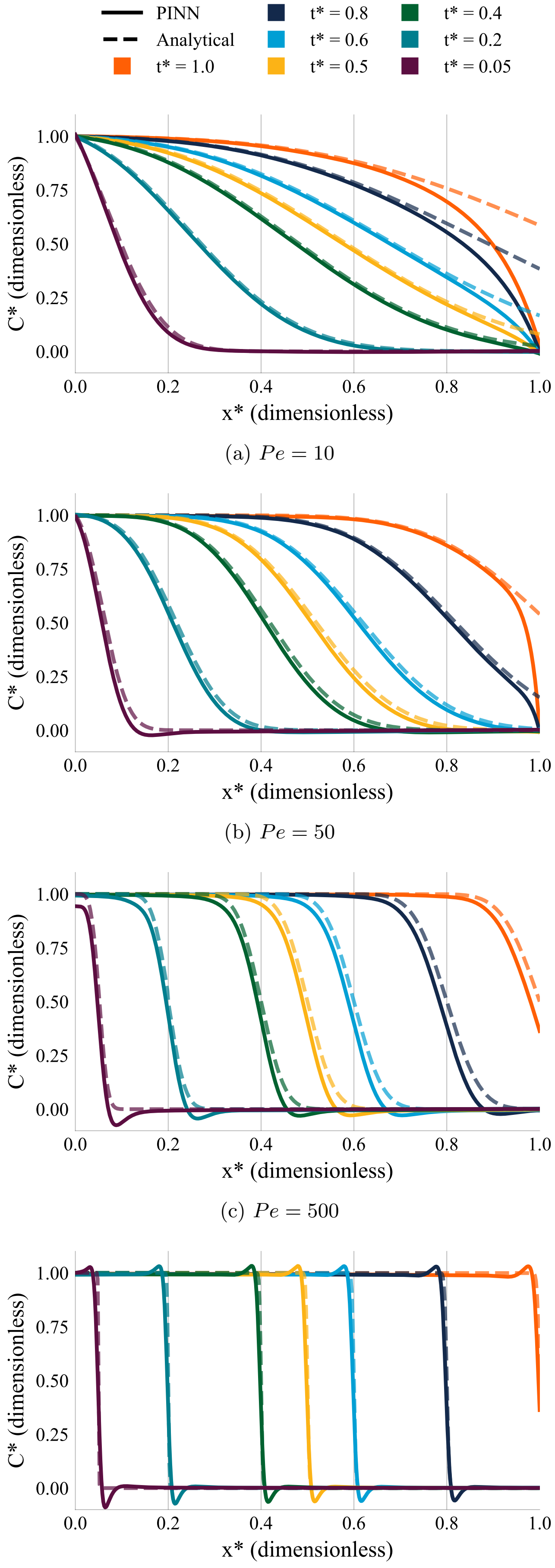
3 Summary and Next Steps

The baseline PINN produces stable dimensionless concentration profiles, and the parametric extension captures the transition from diffusion-dominated to advection-dominated behavior across a wide Peclet range. These results establish a reproducible foundation for parametric transport modeling with PINNs. Future work will shift toward neural-operator approaches.

References

[1] Xavier Glorot and Yoshua Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Vol. 9. Proceedings of Machine Learning Research. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256. URL: https://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf.

[2] Akio Ogata and Robert B. Banks. *A Solution of the Differential Equation of Longitudinal Dispersion in Porous Media*. Professional Paper 411-A. U.S. Geological Survey, 1961. URL: https://pubs.usgs.gov/pp/0411a/report.pdf (visited on 01/01/2025).



(a) $Pe = 10$

(b) $Pe = 50$

(c) $Pe = 500$

(d) $Pe = 10^5$

Figure 3: Parametric PINN C^* profiles across representative Peclet numbers.

A Grid Search for Training Efficiency

To assess the impact of architectural and sampling choices on accuracy and computational cost, a structured grid search over baseline PINN configurations was conducted. This exploratory study was not used directly in the main training scripts but served to inform architecture selection and training design. Each configuration trains a baseline PINN variant, records loss histories, and writes outputs to a unique hash-based directory. Runs are resumable, as configurations with existing saved loss files are automatically skipped. Final losses and wall-clock training times are aggregated into a master CSV file to enable direct comparison of solution quality versus runtime. The full experimental script is available at Experimental/pinn_grid_search.

Table 1 summarizes all evaluated configurations, ranked by final total loss. The results indicate that moderately deep architectures (e.g., three hidden layers with twelve neurons per layer) provide a favorable balance between accuracy and efficiency, consistently outperforming wider single-layer alternatives under the tested settings.

Table 1: Grid search configurations ranked by final total loss.

Rank	Layers	Neurons	Epochs	LR	PDE Collocation	Final Total Loss	Training Time (s)
1	3	12	15000	0.001	2000	0.00286496	84.6
2	3	12	15000	0.001	5000	0.00433964	108.0
3	2	16	15000	0.001	5000	0.00514802	78.8
4	2	16	15000	0.001	2000	0.00539711	63.5
5	3	12	15000	0.001	10000	0.00590407	146.0
6	2	16	15000	0.001	10000	0.00650195	114.6
7	2	16	5000	0.0001	2000	0.00870967	25.2
8	3	12	5000	0.0001	2000	0.01193960	26.3
9	2	16	5000	0.0001	5000	0.01331970	31.9
10	3	12	5000	0.0001	5000	0.01467140	31.9
11	1	64	5000	0.0001	2000	0.15456900	27.0
12	1	64	5000	0.0001	10000	0.16024500	63.0
13	1	64	5000	0.001	2000	0.16057000	27.0
14	1	64	5000	0.001	10000	0.16221000	63.0
15	2	16	5000	0.001	2000	0.16547400	25.2
16	3	12	5000	0.001	2000	0.16603900	26.3
17	2	16	5000	0.001	10000	0.16735700	63.0
18	3	12	5000	0.001	10000	0.16743300	63.0

B Adaptive Collocation with Adam–LBFGS Schedule

This appendix documents an exploratory PINN formulation that combines adaptive collocation with a staged optimization strategy. The model is trained entirely in dimensionless form, mapping $(x^*, t^*) \mapsto C^*$, with conversion to physical units performed only for visualization and comparison. The purpose of this study is to investigate whether targeted collocation refinement and optimizer scheduling can improve training efficiency and solution quality for transport problems. The full experimental script is available at Experimental/pinn_distribution_adaptive_adam_lbfgs.

Adaptive collocation (loss-distribution driven). Rather than using a fixed collocation set, the PDE residual is first evaluated on a uniform (x^*, t^*) grid to estimate a spatial–temporal loss distribution. New collocation points are then sampled using an inverse cumulative distribution function (CDF) constructed from this loss field, so that regions with larger residuals receive proportionally higher sampling density. This approach concentrates training effort in dynamically identified regions of difficulty while preserving a fixed overall collocation budget.

Anchors and adaptive points. To balance global coverage with local refinement, the collocation set is partitioned into two subsets: anchors, which remain fixed throughout training, and adaptive points, which are periodically redistributed according to the loss distribution. The relative fraction of anchors provides a continuous control between purely adaptive sampling and fully uniform sampling, enabling systematic exploration of refinement strategies without sacrificing domain coverage.

Optimizer schedule. Training follows a two-stage optimization procedure consisting of an initial stochastic phase followed by a deterministic refinement phase. The first stage promotes rapid reduction of the loss landscape and robust exploration, while the second stage emphasizes fine-scale convergence once the solution has approached a physically consistent regime. This staged approach is commonly used in PINN training to combine stability with final accuracy.

Algorithm (high level).

1. Initialize a mixed collocation set consisting of anchor and adaptive points.

2. Train the network using a stochastic optimizer while monitoring the PDE residual.

3. Periodically estimate the residual distribution and resample adaptive points via inverse-CDF sampling.

4. Transition to a second-order optimizer for final solution refinement.

Configurable components. Several elements of the procedure are configurable to support exploratory studies, including the frequency of adaptive updates, the balance between fixed and adaptive collocation points, and the relative weighting of loss terms.

Outputs from this framework include dimensionless concentration profiles with analytical overlays, collocation-density visualizations, PDE-residual heatmaps, and loss histories, providing multiple diagnostic views of training behavior and solution quality.